Assignment 16.11.21

1.)Decode a string recursively encoded
as count followed by substring

<count>[sub_str] ==> The substring 'sub_str' appears count times.

Input : str[] = "1[b]"

Output : b

Input : str[] = "2[ab]"

Output : abab

Input : str[] = "2[a2[b]]"

Output : abbabb

Input : str[] = "3[b2[ca]]"

Output : bcacabcacabcaca

PROGRAM :

```
package com.company;
import java.util.Stack;
class Main {
        static String decode(String str) {
            Stack<Integer> integerstack = new Stack<>();
            Stack<Character> stringstack = new Stack<>();
            String temp = "", result = "";
            for (int i = 0; i < str.length(); i++) {
                int count = 0;
                if (Character.isDigit(str.charAt(i))) {
                    while (Character.isDigit(str.charAt(i))) {
                        count = count * 10 + str.charAt(i) - '0';
                        i++;
                    }
                    i--;
                    integerstack.push(count);
                }
                else if (str.charAt(i) == ']') {
                    temp = "";
                    count = 0;
                    if (!integerstack.isEmpty()) {
                        count = integerstack.peek();
                        integerstack.pop();
                    }
                    while (!stringstack.isEmpty() &&
stringstack.peek()!='[' ) {
                        temp = stringstack.peek() + temp;
                        stringstack.pop();
                    }
                    if (!stringstack.empty() && stringstack.peek() == '[')
                        stringstack.pop();
                    for (int j = 0; j < count; j++)
                        result = result + temp;
                    for (int j = 0; j < result.length(); j++)
                        stringstack.push(result.charAt(j));
                    result = "";
                }
                else if (str.charAt(i) == '[') {
                    if (Character.isDigit(str.charAt(i-1)))
                        stringstack.push(str.charAt(i));
                    else {
                        stringstack.push(str.charAt(i));
                        integerstack.push(1);
                    }
                }
                else
                    stringstack.push(str.charAt(i));
            }
            while (!stringstack.isEmpty()) {
                result = stringstack.peek() + result;
                stringstack.pop();
            }
            return result;
        }
        public static void main(String args[]) {
            String str = "3[b2[ca]]";
            System.out.println(decode(str));
        }
}
```

OUTPUT :

2.)Calculate the total fine to be collected
Given a date and an array of integer containing the numbers of the cars traveling on that date(an integer), the task is to calculate the total fine collected based on the following rules:

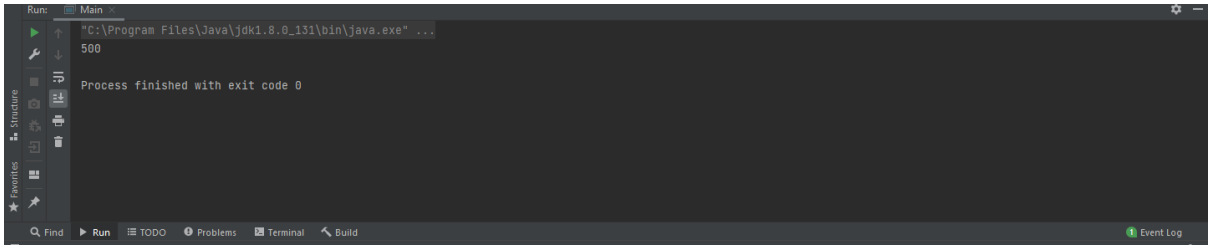
- Odd numbered cars can travel on only odd dates.
- Even numbered cars on only even dates.
- Otherwise a car would be fined 250 Rs.

Examples:
Input: car_num[] = {3, 4, 1, 2}, date = 15
Output: 500
Car with numbers '4' and '2' will be fined 250 each.
Input: car_num[] = {1, 2, 3} , date = 16
Output: 500
Car with numbers '1' and '3' will be fined 250 each.

PROGRAM:

```
package com.company;
class Main {
        static int totFine(int car_num[], int n, int date, int fine) {
            int tot_fine = 0;
            for (int i = 0; i < n; i++)
                if (((car_num[i] ^ date) & 1) == 1)
                    tot_fine += fine;
            return tot_fine;
        }
        public static void main(String[] args) {
            int car_num[] = { 3, 4, 1, 2 };
            int n = car_num.length;
            int date = 15, fine = 250;
            System.out.println(totFine(car_num, n, date, fine));
        }
}
```

OUTPUT :


Run:
Main
"C:\Program Files\Java\jdk1.8.0_131\bin\java.exe" ...
500
Process finished with exit code 0
Structure
Favorites
Find
Run
TODO
Problems
Terminal
Build
Event Log